

Konsep Dasar Desain Responsif

- **Apa itu Desain Responsif?:** Desain yang memungkinkan halaman web menyesuaikan tampilannya sesuai dengan perangkat yang digunakan (desktop, tablet, ponsel).
- **Pentingnya Desain Responsif:** Menjamin aksesibilitas, meningkatkan pengalaman pengguna (UX), dan mempertahankan kesesuaian konten di semua perangkat.

Topik Pembelajaran dan Praktik Responsif dengan Tailwind CSS

1. Breakpoints pada Tailwind CSS

- **Breakpoints yang Digunakan:** sm, md, lg, xl, 2xl untuk menargetkan perangkat dengan ukuran layar yang berbeda.
- **Prinsip Mobile-First:** Mendefinisikan styling dasar untuk perangkat kecil, lalu menambahkan styling khusus untuk perangkat besar dengan menggunakan breakpoints.

Praktik:

a. Mengatur Ukuran Teks Berdasarkan Breakpoints

```
<h1 class="text-lg sm:text-xl md:text-2xl lg:text-3xl xl:text-4xl">
```

Judul Halaman

```
</h1>
```

Penjelasan: Ukuran teks akan berubah berdasarkan breakpoints **sm, md, lg, xl**.

b. Mengubah Layout Menggunakan Flexbox Responsif

```
<div class="flex flex-col md:flex-row">
```

```
<div class="w-full md:w-1/3 bg-gray-200 p-4">Sidebar</div>
```

```
<div class="w-full md:w-2/3 bg-white p-4">Konten Utama</div>
```

```
</div>
```

Penjelasan: Mengubah layout flex dari kolom ke baris pada layar dengan lebar **md** ke atas.

Membuat Tabel Responsif

- Menggunakan Overflow dan Breakpoints untuk Tabel

Praktik:

2. Tabel Responsif dengan Scroll

```
<div class="overflow-x-auto">
  <table class="min-w-full bg-white">
    <thead>
      <tr>
        <th class="py-2 px-4 bg-gray-200">Header 1</th>
        <th class="py-2 px-4 bg-gray-200">Header 2</th>
        <th class="py-2 px-4 bg-gray-200">Header 3</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td class="py-2 px-4">Data 1</td>
        <td class="py-2 px-4">Data 2</td>
        <td class="py-2 px-4">Data 3</td>
      </tr>
      <tr>
        <td class="py-2 px-4">Data 4</td>
        <td class="py-2 px-4">Data 5</td>
        <td class="py-2 px-4">Data 6</td>
      </tr>
    </tbody>
  </table>
</div>
```

Penjelasan: Penggunaan class **overflow-x-auto** memungkinkan tabel untuk di-scroll secara horizontal pada perangkat kecil.

3. Membuat Card Responsif

- Menggunakan Breakpoints untuk Mengatur Ukuran Card
- Praktik:**

Card Responsif dengan Breakpoints

```
<div class="container mx-auto p-4">
  <div class="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 gap-4">
    <div class="bg-white shadow-md rounded-lg p-4">
      <h2 class="text-lg font-bold mb-2">Card Title 1</h2>
      <p class="text-gray-600">Card description goes here.</p>
    </div>
    <div class="bg-white shadow-md rounded-lg p-4">
      <h2 class="text-lg font-bold mb-2">Card Title 2</h2>
      <p class="text-gray-600">Card description goes here.</p>
    </div>
    <div class="bg-white shadow-md rounded-lg p-4">
      <h2 class="text-lg font-bold mb-2">Card Title 3</h2>
      <p class="text-gray-600">Card description goes here.</p>
    </div>
  </div>
</div>
```

Penjelasan: Penggunaan **grid-cols-** berdasarkan breakpoints (sm, md) membuat card terlihat responsif pada berbagai ukuran layar.

4. Membuat Navigasi Responsif

- Menggunakan Breakpoints untuk Mengubah Tampilan Menu Navigasi

Praktik:

- Navigasi Responsif dengan Breakpoints

<!DOCTYPE html>

```

<html lang="id">
<head>
  <meta charset="UTF-8">
  <title>Website Profil</title>
  <!-- Menghubungkan Tailwind CSS -->
  <link
href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tai
lwind.min.css" rel="stylesheet">
  <!-- Menghubungkan Font Awesome -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0-beta3/css/all.min.css">
</head>
<body class="bg-gray-100 font-sans text-gray-800">

<nav class="bg-gray-800 p-4">
  <div class="container mx-auto flex flex-wrap items-center
justify-between">
    <div class="text-white text-lg">Brand</div>

    <!-- Tombol menu untuk perangkat kecil -->
    <div class="block lg:hidden">
      <button class="text-white focus:outline-none">
        <i class="fas fa-bars w-6 h-6"></i> <!-- Ikon menu dari
Font Awesome -->
      </button>
    </div>

    <!-- Navigasi menu untuk perangkat besar -->
    <div class="hidden w-full lg:flex lg:items-center lg:w-auto">
      <ul class="lg:flex lg:space-x-4">
        <li><a href="#" class="text-white">Home</a></li>
        <li><a href="#" class="text-white">About</a></li>
        <li><a href="#" class="text-white">Contact</a></li>
      </ul>
    </div>
  </div>
</div>

```

</nav>

</body>

</html>

Penjelasan: Menu navigasi menggunakan breakpoint **lg** untuk menampilkan menu di layar besar dan tombol hamburger di layar kecil.

Detail Penjelasan:

a. **Deklarasi HTML dan Head:**

<link

href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css" rel="stylesheet">

<link rel="stylesheet"

href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">

- Menghubungkan **Tailwind CSS** untuk mengakses kelas utilitas gaya Tailwind.
- Menghubungkan **Font Awesome** untuk menggunakan ikon, seperti ikon menu bar (fa-bars) di perangkat kecil.

b. **Navigasi Utama:**

<nav class="bg-gray-800 p-4">

- **bg-gray-800:** Menentukan latar belakang navbar berwarna abu-abu tua.
- **p-4:** Memberikan padding di semua sisi sebesar 4 (16px).

c. **Kontainer Navbar:**

<div class="container mx-auto flex flex-wrap items-center justify-between">

- **container:** Membuat batas lebar maksimum untuk memastikan konten tetap terpusat dan responsif.

- `mx-auto`: Memberikan margin horizontal otomatis untuk menempatkan kontainer di tengah layar.
- `flex`: Mengatur elemen dalam kontainer menggunakan flexbox untuk tata letak fleksibel.
- `flex-wrap`: Mengatur agar elemen di dalam kontainer bisa turun ke baris berikutnya jika terlalu panjang.
- `items-center`: Menyelaraskan elemen secara vertikal di tengah kontainer.
- `justify-between`: Memberikan ruang maksimal antar elemen, menempatkan mereka di sisi kiri dan kanan kontainer.

d. **Brand (Teks Brand di Kiri):**

```
<div class="text-white text-lg">Brand</div>
```

- `text-white`: Mengatur warna teks menjadi putih agar kontras dengan latar belakang navbar.
- `text-lg`: Mengatur ukuran teks menjadi lg (18px).

e. **Tombol Menu untuk Perangkat Kecil:**

```
<div class="block lg:hidden">
```

```
  <button class="text-white focus:outline-none">
```

```
    <i class="fas fa-bars w-6 h-6"></i>
```

```
  </button>
```

```
</div>
```

- `block`: Mengatur elemen `<div>` ini sebagai blok agar muncul pada perangkat kecil.
- `lg:hidden`: Mengatur tombol ini menjadi tersembunyi pada layar lg ke atas (min 1024px).
- `text-white`: Memberikan warna putih pada tombol.
- `focus:outline-none`: Menghilangkan outline default saat tombol di-klik atau difokuskan.
- `<i class="fas fa-bars w-6 h-6"></i>`: Menggunakan ikon "bars" dari Font Awesome untuk menu, dengan ukuran w-6 h-6 (24px).

f. **Navigasi Menu untuk Perangkat Besar:**

```
<div class="hidden w-full lg:flex lg:items-center lg:w-auto">
  <ul class="lg:flex lg:space-x-4">
    <li><a href="#" class="text-white">Home</a></li>
    <li><a href="#" class="text-white">About</a></li>
    <li><a href="#" class="text-white">Contact</a></li>
  </ul>
</div>
```

- **hidden:** Menyembunyikan div ini pada layar kecil.
- **lg:flex:** Menampilkan div ini sebagai flexbox pada layar besar (ukuran lg atau lebih besar).
- **lg:items-center:** Menyelaraskan elemen navigasi secara vertikal di tengah.
- **lg:w-auto:** Menyesuaikan lebar div sesuai konten pada layar besar.
- **lg:flex lg:space-x-4:** Membuat daftar dengan flexbox pada layar besar dan memberikan jarak horizontal (space-x-4) antar elemen sebesar 4 (16px).

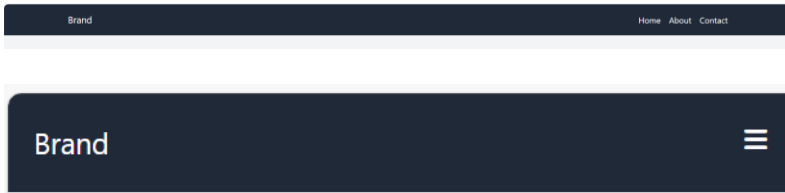
g. **Link Menu:**

```
<li><a href="#" class="text-white">Home</a></li>
```

- **text-white:** Memberikan warna teks putih untuk link, agar terlihat jelas di atas latar belakang abu-abu navbar.

Kode ini menciptakan navbar yang responsif:

- **Perangkat Kecil (Mobile):** Menampilkan ikon menu Font Awesome (ikon "bars") untuk memicu menu pada perangkat kecil.
- **Perangkat Besar:** Menampilkan menu navigasi lengkap (Home, About, Contact) di samping kanan navbar, dengan gaya flexbox yang menempatkan elemen secara horizontal dan rapi.



Gambar 1. Navbar Posisi besar dan kecil

5. Membuat Formulir Responsif

- Menggunakan Tailwind CSS untuk Mengatur Formulir yang Adaptif

Praktik:

- Formulir Responsif dengan Breakpoints

```
<div class="container mx-auto p-4">
  <form class="space-y-4">
    <div class="flex flex-col md:flex-row gap-4">
      <div class="w-full md:w-1/2">
        <label class="block mb-2 text-sm text-gray-600">Nama
        Depan</label>
        <input type="text" class="w-full p-2 border border-gray-300
        rounded-md">
      </div>
      <div class="w-full md:w-1/2">
        <label class="block mb-2 text-sm text-gray-600">Nama
        Belakang</label>
        <input type="text" class="w-full p-2 border border-gray-300
        rounded-md">
      </div>
    </div>
    <div class="w-full">
      <label class="block mb-2 text-sm text-gray-
        600">Email</label>
      <input type="email" class="w-full p-2 border border-gray-
        300 rounded-md">
    </div>
  </form>
</div>
```



```

</div>
<div class="w-full">
  <button type="submit" class="w-full bg-blue-500 text-white
p-2 rounded-md">
    Submit
  </button>
</div>
</form>
</div>

```

Penjelasan:

- space-y-4 menambahkan margin bottom sebesar 16px pada setiap elemen di dalam kontainer, kecuali elemen terakhir.

Contoh:

```

<div class="space-y-4"> <div class="bg-blue-500 p-4">Elemen 1</div> <div class="bg-green-500 p-4">Elemen 2</div> <div class="bg-red-500 p-4">Elemen 3</div> </div>

```

Pada contoh di atas, setiap elemen memiliki jarak vertikal sebesar 16px antara satu sama lain, berkat penggunaan space-y-4.

- Kelas md:w-1/2 dalam Tailwind CSS adalah kelas responsif yang mengatur lebar elemen menjadi setengah dari kontainer (1/2 atau 50%) pada layar dengan ukuran minimal **medium** (md) atau lebih besar.

Penjelasan Detail

- **md::** Ini adalah prefix yang mengindikasikan bahwa gaya ini hanya akan berlaku pada layar ukuran medium atau lebih besar, yaitu **minimal 768px** lebar layar.
- **w-1/2:** Ini adalah utility class di Tailwind untuk menetapkan lebar elemen sebesar 50% dari kontainer atau elemen induknya.
- **w-1/3:** Mengatur lebar elemen menjadi **sepertiga (33.33%)** dari lebar kontainer.

- **w-1/4**: Mengatur lebar elemen menjadi **seperempat (25%)** dari lebar container.

Contoh Penggunaan

Misalnya, Anda memiliki dua elemen dalam sebuah div, dan ingin keduanya memiliki lebar 50% pada layar medium ke atas (768px atau lebih), tetapi menampilkan satu per satu (lebar 100%) pada layar yang lebih kecil dari itu. Anda bisa menggunakan md:w-1/2 untuk mengatur gaya tersebut.

```
<div class="flex flex-wrap">
  <div class="w-full md:w-1/2 bg-blue-500 p-4">Konten 1</div>
  <div class="w-full md:w-1/2 bg-green-500 p-4">Konten
2</div>
</div>
```

Hasilnya:

Nama Depan	Nama Belakang
<input type="text"/>	<input type="text"/>
Email	
<input type="text"/>	
<input type="submit" value="Submit"/>	

Gambar 2. Formulir Responsif

1. Tugas Membuat Formulir seperti dibawah berikut:

Formulir Pendaftaran

Nama Lengkap

Email

Telepon

Kota

Provinsi

Kode Pos

Daftar

Gambar 3. Tugas Formulir

2. Tugas Card 1



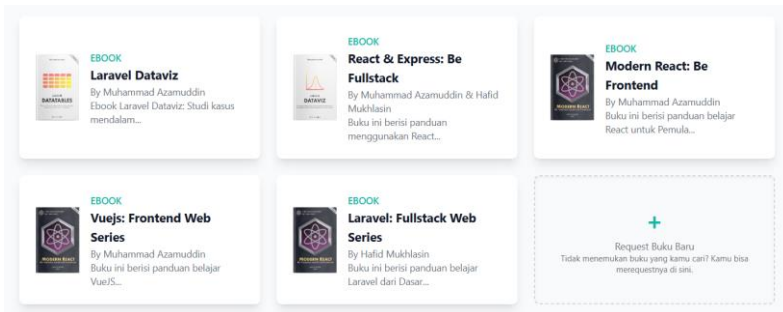
Gambar 4. Tugas Card 1

3. Tugas Card 2



Gambar 5. Tugas Card 2

4. Tugas Card 3



Gambar 6. Tugas Card 3

5. Tugas Card 4



Gambar 7. Tugas Card 4

